Integrating Parallel and Distributed Computing in Early CS Courses

Sheikh Ghafoor, Tennessee Tech University Sushil Prasad, University of Texas San Antonio Charles Weems, University of Massachusettes Contact: sghafoor@tntech.edu

Parallel and distributed computing (PDC) has become pervasive in all aspects of computing, and thus it is essential that students include parallelism and distribution in the computational thinking that they apply to problem solving, from the very beginning. Computer science education is still teaching to a 20th century model of algorithmic problem solving. Sequence, branch, and loop are taught in our early courses as the only organizing principles needed for algorithms, and we invest considerable time in showing how best to sequentially process large volumes of data. All computing devices that students use currently have multiple cores as well as GPU in many cases. Most of their favorite applications use multiple cores and numbers of distributed processors. Often concurrency offers simpler solutions than sequential approaches. ACM and ABET have recommended including PDC in the undergraduate CS curriculum. However, we are still teaching them to solve problems using sequential thinking. In this workshop we overview the key PDC concepts and provide examples of how they may naturally be incorporated in early CS classes. We will introduce plugged and unplugged curriculum modules that have been successfully integrated in existing CS classes at multiple institutions. We will highlight the upcoming summer training that we are organizing, for which we have funding to support attendance.

Keywords: PDC Education; Early Computing Class; HPC Education

DOI: <u>HTTPS://DOI.ORG/10.1145/3478432.3499155</u>